Automated Molecular Mechanics Parameterization with Simultaneous Utilization of Experimental and Quantum Mechanical Data

PER-OLA NORRBY, TOMMY LILJEFORS

Department of Medicinal Chemistry, The Royal Danish School of Pharmacy, Universitetsparken 2, DK-2100 Copenhagen, Denmark

Received 2 June 1997; accepted 20 October 1997

ABSTRACT: A general set of procedures for automated parameterization of molecular mechanics force fields is presented. The current implementation for MacroModel force fields can easily be adapted to other programs running under Unix. Several variants of numerical Newton–Raphson and simplex methodologies are used to fit simultaneously both experimental and *quantum mechanical* data. In a sample parameterization of ethane, 24 different modes of combining various sources of reference data are compared. The best cost/performance ratio for generating quantum mechanical source data was obtained with the B3LYP/6-31G* method. The best optimization strategy consists of initial subset optimizations with a modified simplex method, followed by Newton–Raphson optimization using Lagrange multipliers. © 1998 John Wiley & Sons, Inc. J Comput Chem 19: 1146–1166, 1998

Keywords: automatic parameterization; force field development; molecular mechanics; MM3; Newton–Raphson and simplex optimization

Introduction

During the last decades, analysis of molecular structure and energetics by force field methods[1] (molecular mechanics and molecular dynamics) has developed from a specialist's tool to a general methodology that can be easily utilized in many fields with only a basic knowledge of the underlying techniques. A variety of previously inaccessible problems can now be solved by structure and energetics analysis. However, the widespread utilization also frequently results in abuse of the methodology. The ease of use of current commercial packages leads to their fre-

Correspondence to: P.-O. Norrby, e-mail; peon@medchem.dfh.dk

Contract/grant sponsors: The Danish Medical Research Council; Lundbeck Foundation, Copenhagen

CCC 0192-8651 / 98 / 101146-21

quent application to problems in which the underlying force field methods are ill-defined or, worse, undefined.

A substantial part of the erroneous or dubious use of force field methodology lies in the confusion of various types of energy (e.g., free energy vs. enthalpy vs. heat of formation vs. steric energy, or use of gas phase calculations for strongly solvated systems), but an equally serious abuse lies in the utilization of low-quality parameters. One of the major obstacles in the introduction of the molecular mechanics methodology to nonspecialists is the lack of high-quality parameters for many systems. In practice, almost any new system will require parameters that have not previously been determined accurately. The solution in many force field packages has been to introduce ''general'' or ''automatic'' parameters. These parameters can be very useful in that they allow rapid evaluation of widely differing systems, but it must be realized that high-quality results require high-quality parameters.

Molecular mechanics parameterization has long been more of an art than a science.[2] The problem is twofold: finding and evaluation of relevant data, and fitting of the parameters. The former problem is complicated by the fact that most molecular mechanics methods are basically gas phase tools, and thus require structural and energetic data from gas phase determinations. The latter problem requires an optimization tool (frequently a least-squares minimization method),[3–9] but also a weighting of the input data depending on both data quality and type, and on the objectives of the force field (e.g., structural prediction, energetic comparisons, or calculation of vibrational spectra).

For many types of structures (in particular those incorporating transition metals) it is almost impossible to obtain the required gas phase data for parameterization. Commonly, a database of crystallographic structures and solution energetics has been used in lieu of gas phase data, with the implicit assumption that condensed phase effects can be neglected if an average is taken over a large enough data set. In some cases, crystal packing effects have been calculated and used explicitly in the parameterization procedure.[3–5,9] During the last decades, quantum mechanical calculations have increasingly been used to extend the available data set.[6–8,10–13] The results from well-parameterized molecular mechanics calculations are still substantially more accurate than production-level quantum mechanical (QM) calculations (*vide infra*), and it is necessary to weigh the quantum mechanical data versus the relatively scarce experimental data so as not to lose the high accuracy that can be obtained. Several paradigms have been advanced for the introduction of quantum mechanical data in the parameterization procedure. Hopfinger and Pearlstein[10] derived a method for fitting molecular mechanics parameters to QM-calculated energies for distorted structures, assuming that the best possible result for the molecular mechanics method would be to fit the QM energies as close as possible. Palmö et al.[6] divided the molecular mechanics energy into quadratic and nonquadratic parts, assumed the nonquadratic part to be determinable from other sources, and fitted the quadratic part exactly to a QM-derived Cartesian Hessian (a matrix of energy second derivatives).

Hagler et al.[8] used a two-step procedure. First, a QM force field was parameterized entirely based upon production calculations at the relatively low level HF/6-31G*. The CFF force field was then determined from the QM force field by the introduction of a small number of scale factors that could be reliably determined from fitting to the experimental data set. CFF includes a large number of cross-terms to enable an accurate description of high-energy nonstationary points on the potential energy hypersurface (necessary for very distorted structures and for the accurate calculation of vibrational data). A reliable set of cross-terms is very hard to achieve by fitting to experimental data only, but the necessary balance was obtained by the utilization of energy second derivatives (the Cartesian Hessian) in the parameterization of the QM force field. An added advantage is the very large number of data points obtained from the quantum mechanical Hessian. By inclusion of several distorted structures in the parameterization procedure, the investigators obtained over 1900 independent data points for each parameter to be determined.

Halgren[7] used the method of Hagler et al.[8] for fitting the parameters of MMFF to quantum mechanical data. However, instead of relying on parameter scaling to fit experimental data, Halgren instead used increasing levels of theory for different types of quantum mechanical predictions, with the objective to achieve experimental accuracy in the data before using it in the parameterization procedure. The basis of the parameterization is a

set of energies and structures obtained at the HF/6-31G* and MP2/6-311G** levels, but, particularly in the determination of parameters for nonbonded interactions, substantially higher levels of theory were utilized.

Several other groups have utilized quantum mechanical data in force field parameterization, but in many cases the specific advantage of being able to calculate nonstationary points on the potential energy hypersurface (PES) has not been utilized. The quantum mechanical calculations have more commonly been employed to determine properties corresponding to experimental observables (e.g., structures or rotational barriers), which have been used in lieu of observed data. Norrby et al.[13] developed a method for rapid determination of missing torsional parameters within the context of an existing force field from computationally cheap single-point quantum mechanical calculations at nonstationary points. The method, which can be viewed as a small subset of the procedures of Hagler et al.,[8] was shown to give an accuracy comparable to the more demanding determination of a fully relaxed rotational barrier at a much lower cost.

Both the paradigms of Hagler et al.[8] and Halgren[7] have been shown to yield force fields that are among the most reliable currently available.[14] The current work will build upon their methodologies, but will follow a slightly different paradigm. The emphasis will be on the concept of adding parameters to an existing force field, and also to utilize combinations of suitably weighted experimental and quantum mechanical data to achieve both the necessary balance between parameters and the desired accuracy in the fit to the experimental data set. The final goal is to make generally available a nonproprietary set of procedures for automation of molecular mechanics parameterization.

The methodology described herein is currently being used in parameterization of several types of force fields, among them models of new heterocyclic systems, transition states, and organometallic systems. The specific problems of parameterizing the more unusual systems will be detailed elsewhere. The detailed example used in the current work is taken from the most well-known area in molecular mechanics, the simple alkanes. Several complete reparameterizations of ethane will be used to demonstrate the effect of different parameterization methodologies.

Overview

Determination of a new set of force field parameters consists of several distinct steps:

- Collection of reference data and definition of a merit function. Usually, the merit function is a weighted sum of squares of deviations between reference data points and the corresponding calculated force field values.
- Definition of a functional form. Even within the context of an existing force field, parameterization of new functional groups commonly requires definition of what atom types and bond types to use.
- Choice of initial parameter values.
- Refinement of parameters (optimization of the merit function).
- Testing and validation of the final parameter set.

The main subject of the current work is parameter refinement, which is well suited for automation. However, no optimization is possible without first determining an unoptimized but functioning set of parameters. Therefore, an overview of the initialization procedure will be given (*vide infra*).

FORCE FIELD

The set of procedures in the current work can be applied to any type of force field parameterization, but the specific implementation described here is developed for the force fields in MacroModel,[15] and in particular MM3*. The current implementation is in the form of Unix shell scripts and short C-programs, and can be applied as is to any MacroModel force field. The scripts depend on the specific text format in MacroModel output and force field files, but can be easily modified to work with any program running under Unix. The choice of program and force field is based on several criteria: (1) The program can be executed in batch under Unix, and is therefore well suited for automation. (2) Both output and force fields are contained in consistently formatted text files, which are easily manipulated. (3) The force fields available in MacroModel are among the most accurate currently available,[14] and are therefore suitable bases for addition of subsets of parameters. (4) MacroModel performs calculations with ''general''

parameters when needed, but reports all low-quality parameters utilized, and thus quickly identifies situations where additional parameterization is required. (5) The MM3* force field uses a functional form that enables it to reproduce vibrational spectra with fair accuracy, a distinct advantage when utilizing a quantum mechanically determined Hessian in the parameterization.

The MM3* force field is based on, but not identical to, the 1989 version of the Allinger MM3 force field.[9] The parameters are usually interchangeable, but some differences in paradigm should be noted. The electrostatics in MM3 are based mainly on dipole interactions, whereas MM3* utilizes point charges determined by using the MM3 dipoles as charge flux parameters. The two schemes usually give similar results, but reparameterization of MM3* may be necessary for systems with proximal, strongly polar functional groups. MM3* also uses a nondirectional 10–12 Lennard–Jones-type potential for hydrogen bonds in lieu of the directional Buckingham potential in MM3. A more serious discrepancy lies in the treatment of conjugated systems. MM3 uses a VESCF scheme to determine π-bond orders, and scales bond and torsional parameters according to the calculated bond order. MM3* instead relies on user identification of single or double bonds, and uses a substructure matching scheme to identify and parameterize specific conjugated systems. In several cases, this leads to different calculated results for different input Kekule forms of conjugated molecules,[14] and will definitely require determination of new parameters for many conjugated systems.

DEFINITION OF PARAMETERIZATION OBJECTIVE

In the automated parameterization procedure, the overall performance of the force field is defined as a merit function, χ^2, a sum of squares of all deviations of calculated values, $\hat{y}_i$, from the reference values, y_i° [eq. (1)]:

$$\chi^2 = \sum_i w_i^2 (y_i^\circ - \hat{y}_i)^2 \qquad (1)$$

The best set of parameters is that which minimizes χ^2. The deviations in eq. (1) are scaled by the weighting factors, w_i making the expression dimensionless. The choice of weighting factors is an important step in parameterization, and should be influenced by the expected future usage of the force field. For example, the force field can be biased toward accurate reproduction of structures or, alternatively, by giving high weights to energy second derivatives, toward reproducing vibrational spectra. The weighting factors should also be influenced by the relative amount of different kinds of data to be used in the parameterization (e.g., there may be several magnitudes more energy derivatives than relative energies available). It is also possible to vary the weighting by the quality of the reference data (e.g., according to standard deviations in x-ray structures). In the current work, the weighting factors have only been differentiated according to type of reference data. The values in Table I are used throughout the current work.

TABLE I.
Weighting Factors.

Type of data	w_i
Bond length	100 Å^{-1}
Bond angle	2 degree^{-1}
Dihedral angle	1 degree^{-1}
Relative energy	100 kJ^{-1} mol
Mass-weighted energy second derivative	0.01 kJ^{-1} mol Å^2 amu^{-1}

Reference Data

The first part of the initialization procedure consists of selecting the data to be used in the parameterization and setting up the automatic system for calculating each data point using the current force field. The heart of the parameterization procedure is to calculate each input value automatically and to compare it with the reference value. The details on how to accomplish this will be discussed separately for each type of input data. The common procedure for all types of data in the current implementation is to generate a set of BatchMin command and data files,[16] then, in each iteration, execute the files and extract the relevant data from the output.

STRUCTURAL DATA

The most common type of structural data in parameterization comes from x-ray crystallographic determinations. In the MM3 paradigm,

minimized structures correspond to gas phase structures as determined by electron diffraction. The small discrepancy between the structural types will be ignored in the current work, except to note that hydrogen positions may not be determined from x-ray data.

There are many ways that a molecular structure can be utilized in parameterization. The conceptually simplest method would be to calculate the Cartesian rms deviation obtained from a least-squares superposition of the reference and minimized structures. This procedure has two obvious drawbacks. A minor drawback is that least-squares superposition cannot easily be automated in the current version of MacroModel. A major drawback is that the Cartesian deviation is not a good measure of the quality of the force field. Soft distortion modes are commonly affected strongly by crystal packing, with the result that the gas phase minimum may differ substantially from the observed crystal structure. Thus, a better measure would be to calculate the steric energy of the crystal structure and compare it with the steric energy of the corresponding minimized structure. This is a good way of validating the final force field, but the drawback of this measure in the parameterization cycle is that it will tend to give low values to all force constants. In the extreme case, setting all force-related constants to zero will give a zero energetic deviation between calculated and reference structure, and thus the impression of an optimum fit! The same is true for the method of calculating the Cartesian forces on the atoms in the reference structure and trying to minimize these forces. We have instead chosen the method of treating each bond length, bond angle, and dihedral angle for which a parameter is being refined as a data point in the parameterization. It would be possible to include all structural elements from all structures as data points, but those for which no parameter is modified will surely be quite insensitive to the parameter changes, and thus contribute little to the refinement. Adding unneeded structural elements also increases the risk that spurious correlations between errors in the structures and random parameters introduce instabilities in the parameterization procedure.

ENERGY DATA

The energies most suitable for utilization in a parameterization scheme are relative potential energies or enthalpies, either for different conformations of the same molecule, or for rotational barriers. Enthalpies mainly differ from potential energies by a vibrational contribution that can usually be neglected for relative energies between conformations. However, experimental data are more commonly available in the form of free energies. These are sometimes compared directly to steric energies calculated by force field, but the error due to neglect of the entropic contribution may be substantial. Two main methods are available for inclusion of free energy data in the parameterization. The experimental free energies can sometimes be converted to reasonably accurate gas phase enthalpies by estimation of the entropic contribution and possibly the solvation terms. These contributions may be estimated from preliminary force field calculations and updated during parameterization. Alternatively, with force fields that are able to calculate thermodynamic contributions from normal mode analysis (e.g., MM3), the calculated free energy can in each iteration be directly compared with the experimental value. The advantage of the latter method is consistency in the parameterization, as comparison is always made between the same type of data. However, the direct calculation of free energies in each iteration cycle may well require several thousand expensive normal mode analyses. The former method has a distinct speed advantage. It may even be feasible to evaluate the difference between steric and free energy with a more expensive method, like free energy perturbation, because the evaluation need only be corrected a few times in the total parameterization cycle.

It must also be realized that experimentally determined relative energies frequently refer to the overall energy difference between two conformational ensembles. In some cases, it might be sufficient to compare just the lowest energy representatives from each ensemble, but a more general method is to calculate the total ratio from Boltzmann populations for all contributing conformations [eq. (2)]. This is especially important if it is not known *a priori* which conformer is dominant within an ensemble. Note that, in eq. (2), the approximate Boltzmann populations are based on calculated steric energies, not free energies:

$$\Delta E \approx -RT \ln\left(\frac{\sum_i e^{-E_i/RT}}{\sum_j e^{-E_j/RT}}\right) \qquad (2)$$

Energy differences obtained from quantum mechanical calculations, for example, should be used

directly, without Boltzmann averaging, as these are, by definition, potential energy differences between single, well defined conformations.

ENERGY DERIVATIVES

It has been shown recently that Cartesian first and second derivatives calculated by quantum mechanical methods can be very efficiently utilized in parameterization.[6–8] By inclusion of calculated results for nonstationary points, the low-energy portion of the PES can be reliably reproduced by the force field. We have introduced some slight modifications in our implementation. One difference between MM3 and the force fields that have previously been parameterized from calculated energy derivatives is that bond lengths in MM3 correspond to r_g, not r_e values (i.e., gas phase electron diffraction values including an anharmonic vibrational elongation, not the distances between PES minima). Thus, forces calculated in MM3 at a quantum mechanical stationary point will always be nonzero, thereby destabilizing the parameterization procedure. Two solutions are possible: either to exclude energy first derivatives from the parameterization, or to reparameterize MM3 to fit r_e values. We implement the former, in part because outputting forces from MacroModel is nontrivial (*vide infra*).

The energy second derivatives (the Hessian) are less sensitive than the forces to small discrepancies in reference bond length. MM3 does include anharmonicity terms and this changes the Hessian slightly with changes in the reference bond length, but we have chosen to ignore this small effect. The Hessian is evaluated for identical structures in quantum mechanical and force field calculations. It would be possible to restrict the use of Hessians to stationary points (minima and transition states) in the parameterization, and thus avoid the discrepancies in reference bond length by optimizing the structures to a force-field stationary point before each Hessian evaluation. This may be the optimal procedure in special cases, but it has two drawbacks in the general case. First, the shape of the PES at large distances from any stationary point can only be well defined if such nonstationary points are included in the parameterization. Second, in the initial stages of a parameterization, the position of the stationary points may differ widely between the quantum mechanical and force field methods, and thus greatly increase the risk of finding a bad local optimum for the parameters.

The force field Hessian as such could not be obtained due to a serious paradigm discrepancy in MacroModel (see Implementation Details subsection). However, a numerical mass-weighted Hessian could be output. For reasons of expediency, the parameterization therefore uses mass-weighted energy second derivatives in lieu of the true Hessian.

OTHER TYPES OF DATA

Almost any type of data that can be predicted from force field calculations can also be utilized in the parameterization procedure. For example, Allinger et al. used heats of sublimation to refine the van der Waals parameters in the MM2/MM3/MM4 force fields.[9] The electrostatic parameters can be fit to experimental dipole measurements, or to QM calculations.[12] However, in the current work, we have completely avoided refining the nonbonded parameters. In particular, the van der Waals parameters are fundamental to everything else in the force field, and any modification of these will affect not only the systems under investigation but also the performance in widely different areas. In general, we endeavor to modify only parameters that are specific to systems in which accurate parameters have not yet been determined. However, the principle of changing these basic parameters is not different from other types of parameters and, in special cases, like the creation of an entirely new force field,[3–5,7–9] they may be included in the refinement procedure.

Vibrational data are frequently used in the determination of molecular mechanics force constants. With several current force fields, the vibrational frequencies can be calculated and therefore utilized in the parameterization. However, many frequencies are extremely sensitive to small changes in the force field, and the problem of identifying a calculated frequency with the corresponding reference value can prohibitive, particularly in an automated procedure. In cases where the experimental spectrum can be fully interpreted and a complete set of eigenvectors and eigenvalues generated, the mass-weighted Hessian can be calculated and utilized in the parameterization, as described in the Energy Derivatives subsection. For cases where a few frequencies have been identified, for example by isotropic substitution, manual force constant modification is recommended, at least in the initial stage of refinement.

Force Field Initialization

With the reference data in hand, a functioning force field must be defined that can calculate values for each data point. This requires both a definition of functional form and an initial estimate of all parameters.

When the problem at hand is refinement of a few low-quality parameters, the functional form may already be defined, but the introduction of a new functional group sometimes requires decisions about which atom types and bond orders to use, and which already existing parameters may require reparameterization. As an example, we are currently interested in metal pyridine complexes. Here, we must first decide what type of nitrogen to use (neutral or cationic sp^2 N), what type of bond to use between nitrogen and the metal (a single bond or the MacroModel ''zero-order'' coordination bond), and how the angles around the metal should be treated (valence angles or Urey–Bradley interactions[1] between ligands). Decisions about the bonding and atom type scheme are irrevocable in the sense that a later change will require a complete reparameterization for the new bonding situation. Other choices that must be made in the initialization phase include which existing parameters to keep unchanged and which to refine. This type of choice is less definite. It is entirely possible, even recommended, to keep most existing values in the first cycles of the parameterization, and then decide later whether the intermediate results motivate a further refinement of more parameters.

The choice of electrostatic scheme to use must also be made in the initial phase. This is one of the most difficult parts of the introduction of new functional groups in the force field. MacroModel uses atomic point charges, but the charge is not an experimental observable, and thus cannot be parameterized directly. For force fields like AMBER, a well-defined method exists for obtaining the required charges from *ab initio* calculations.[17] The MM2/MM3/MM4 family of force fields has relied instead on experimental dipoles, but these are not always available for the functional groups of interest and, more importantly, the molecular dipole usually does not give a unique solution for the bond dipoles. Additional problems with this approach arise for symmetric or charged systems. We tend instead to use CHELPG charges[18] determined from high-level QM calculations, with a few modifications to fit the MM3 paradigm. A pragmatic approach is to use the same level of calculation for a parameterized system and determine scale factors by comparing the calculated with existing charges. We also set the charge of hydrogens attached to saturated carbon to zero, and adjust the carbon charge to give a constant charge sum. To the extent possible, we then try to reproduce the modified CHELPG charges by adjusting MM3* dipoles, without adding explicit charges. In our experience, this methodology gives reliable results as long as the electrostatic scheme is fixed before actual parameter refinement is initiated. On the other hand, attempts to fit the dipoles to secondary data, like conformational energies, gives rise to oscillations in the parameters and final parmeter sets with low predictive power.

As the final step of the initialization phase, reasonable starting values must be assigned to all new parameters. As the parameters will be refined, the initial guesses may not seem important, but it is entirely possible that the later refinement procedure will find a local optimum for the parameter set that is different from the best possible set of parameters (this is similar to finding a high energy local minimum in a conformational search). Thus, a general knowledge of reasonable values for ''similar'' parameters in the force field can be very valuable. For MM3, the problem of finding initial values, especially in the field of organometallic molecular mechanics, was greatly simplified with the introduction of the ''parameter estimator'' in the 1994 version of the force field. The underlying methods have been published,[19] and can easily be applied to the generation of initial parameters for MM3*.

Parameter Optimization

Two distinct types of parameter optimization techniques are described here, based on multidimensional Newton–Raphson methodology and simplex optimization, respectively.[20]

NEWTON–RAPHSON OPTIMIZATION

Iterative refinement with the multidimensional Newton–Raphson technique requires determination of the first and second derivatives of the merit

function χ^2 with respect to each parameter p:

$$\frac{\partial \chi^2}{\partial p_j} = -2\sum_i w_i^2 (y_i^\circ - \hat{y}_i)\frac{\partial \hat{y}_i}{\partial p_j} \tag{3}$$

$$\frac{\partial^2 \chi^2}{\partial p_j\, \partial p_k} = 2\sum_i w_i^2 \left(\frac{\partial \hat{y}_i}{\partial p_j}\frac{\partial \hat{y}_i}{\partial p_k} - (y_i^\circ - \hat{y}_i)\frac{\partial^2 \hat{y}_i}{\partial p_j\, \partial p_k} \right)$$

$$\approx 2\sum_i w_i^2 \frac{\partial \hat{y}_i}{\partial p_j}\frac{\partial \hat{y}_i}{\partial p_k} \tag{4}$$

In eq. (4), the term containing the second derivative of the calculated data point $\hat{y}_i$ with respect to the parameters can be neglected.[20] We can now calculate iterative refinements Δp_j to the parameters by solving the set of linear equations given in eq. (5)[20]:

$$-\sum_j \frac{\partial^2 \chi^2}{\partial p_j\, \partial p_k}\Delta p_j = \frac{\partial \chi^2}{\partial p_k} \tag{5}$$

Eq. (5) can be more conveniently rewritten in matrix notation using the weighted Jacobian matrix $\mathbf{J}$, the residual vector $\bar{\mathbf{r}}$, and the vector of parameter increments $\bar{\mathbf{x}}$ [eq. (6)]:

$$\mathbf{J}^T\mathbf{J}\bar{\mathbf{x}} = \mathbf{J}^T\bar{\mathbf{r}}, \qquad \mathbf{J} = \begin{pmatrix} w_1\frac{\partial \hat{y}_1}{\partial p_1} & w_1\frac{\partial \hat{y}_1}{\partial p_2} & \cdots \\ w_2\frac{\partial \hat{y}_2}{\partial p_1} & w_2\frac{\partial \hat{y}_2}{\partial p_2} & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

$$\bar{\mathbf{r}} = \begin{pmatrix} w_1(y_1^\circ - \hat{y}_1) \\ w_2(y_2^\circ - \hat{y}_2) \\ \vdots \end{pmatrix} \qquad \bar{\mathbf{x}} = \begin{pmatrix} \Delta p_1 \\ \Delta p_2 \\ \vdots \end{pmatrix} \tag{6}$$

The parameter refinements can now be calculated after a matrix inversion, eq. (7):

$$\bar{\mathbf{x}} = \mathbf{A}^{-1}(\mathbf{J}^T\bar{\mathbf{r}}), \qquad \mathbf{A} \equiv \mathbf{J}^T\mathbf{J} \tag{7}$$

Ideally, the derivatives of each calculated data point with respect to each parameter ($\mathbf{J}$) should be calculated analytically for optimum performance. However, a true analytical expression usually cannot be obtained, as the geometry at which the observable should be calculated is dependent on the parameters. Warshel and Lifson,[3] and later Rasmussen et al.[5] and Dillen,[4] used truncated chain rule expansions to calculate approximate derivatives. Lagrange multipliers were used to avoid extrapolating the parameters beyond the valid region of the truncation. Hagler et al.,[8] on the other hand, based the parameterization of a QM force field entirely upon fixed-geometry *ab initio* calculations. As the force field was parameterized for fixed geometries only, analytical derivatives with respect to the parameters could be determined and utilized in the Newton–Raphson procedure.

We have based the parameterization on numerically determined derivatives [eq. (8)]. The drawback of numerically determined derivatives is the time-consuming calculation. The main advantages are the ease of implementation and the generality. It is also possible to improve the balance between parameters by adjusting the step size in the differentiation:

$$\frac{\partial \hat{y}_i}{\partial p_j} \approx \frac{\hat{y}_i(\bar{\mathbf{p}} + \delta p_j) - \hat{y}_i(\bar{\mathbf{p}})}{\delta p_j}$$

(forward differentiation)

$$\frac{\partial \hat{y}_i}{\partial p_j} \approx \frac{\hat{y}_i(\bar{\mathbf{p}} + \delta p_j) - \hat{y}_i(\bar{\mathbf{p}} - \delta p_j)}{2\delta p_j}$$

(central differentiation) (8)

In practice, the direct matrix inversion used in eq. (7) usually does not yield the best possible solution, as the parameters are frequently strongly correlated. Approximate solutions to eq. (7) are instead generated by singular value decomposition (SVD), by the Levenberg–Marquardt method (L-M), or by addition of Lagrange multipliers. All methods can be seen as modifying the square matrix $\mathbf{A} \equiv \mathbf{J}^T\mathbf{J}$ to create a better-conditioned and more easily inverted form, denoted $\mathbf{A}'$. Several trial solutions are generated simultaneously from different modified matrices $\mathbf{A}'$. All solutions requiring a very large or very small total parameter change are discarded, as are solutions giving negative values to parameters that have been constrained to be positive. The remaining solutions are used to calculate new merit functions, and the best solution retained as a starting point for the next iteration.

SVD decomposes $\mathbf{A}$ into sets of orthogonal vectors that span the complete range of $\mathbf{A}$, and corresponding singular values.[20] An approximation to $\mathbf{A}$ can be constructed by dropping all singular values below a certain threshold, corresponding to near singularities in $\mathbf{A}$. The inverse of the approximate matrix $\mathbf{A}'$ can easily be constructed from simple inverses of the singular values and the

transpositions of the orthogonal vector matrices.[21] In general, small singular values correspond to large parameter changes with a relatively small predicted gain in the merit function.

In the Levenberg–Marquardt method, the matrix **A** is modified to a more diagonal heavy form by multiplication of the diagonal elements by a constant factor $1 + \lambda$, where λ is a positive number.[20] At large λ values, the method is functionally equivalent to a steepest descent method. In the original method, λ is varied depending on the success of previous iterations. In the current implementation, several trial solutions, based on different λ values, are generated and tested in each iteration cycle.

Near singularities in **A** generally result in a precariously balanced, opposite modifications of nearly linearly dependent parameters. The intuitive solution to this problem is to introduce the added requirement that parameter modifications should be as small as possible in the least-squares solution scheme. This is most easily done by adding a set of Lagrange multiplier constraints to the equation system to be solved [eq. (9)]:

$$\xi \Delta p_j = 0 \tag{9}$$

The procedure is implemented by adding the Lagrange multiplier ξ^2 to each diagonal element in **A**. The procedure has the effect of removing linear dependencies in **A** very efficiently even for modest values of ξ.

As mentioned previously, the Jacobian matrix **J** is calculated by numerical differentiation, either as a simple forward differentiation or, at twice the computational cost, as central differences. The use of central differentiation also enables an estimate of the absolute first and second derivatives of the merit function, $d\chi^2/dp_j$ and $d^2\chi^2/dp_j^2$. When close to an optimum parameter set, the second derivatives should all be positive and large compared to the corresponding first derivatives. Single parameter modifications can also be calculated from the one-dimensional Newton–Raphson using these derivatives.

PARAMETER REFINEMENT USING THE SIMPLEX METHOD

One optimization method that avoids the costly calculations of numerical derivatives is the well-known simplex method.[20] The method relies on generating a simplex, a convex geometrical object in parameter space with one more vertex than the total number of parameters to be refined. The vertices correspond to linearly independent sets of parameters. The simplex is moved through parameter space by successive reflections of the worst data point through the average of the remaining points. Depending on the function value in the new point, the simplex may be expanded or contracted.

The standard simplex method is not expected to perform well for systems with more than ca. 10 parameters, due to the number of reorientations required when the number of vertices grow. Real applications usually involve simultaneous refinement of substantially higher numbers of parameters. The current implementation includes a locally developed modification of the standard method, a weighted simplex involving a bias of the reflection point toward the best point(s) (see Appendix). The modified method has been tested with up to ca. 40 parameters, and has shown a more rapid initial convergence than the aforementioned Newton–Raphson-type methods with numerical differentiation. It is therefore well suited for initial refinement of medium-sized parameter sets, as well as optimization of troublesome subsets of larger parameter sets. It has been particularly useful for subset optimization in cases where the central differentiation identifies parameters for which the absolute second derivatives of the merit function ($d^2\chi^2/dp_j^2$) are small or negative.

PARAMETERIZATION STRATEGY

Early on, it was observed that a simple application of the automated parameterization to an interdependent parameter set could yield false optima (corresponding to local minima in conformational searches). Usually, such a local parameter optimum could be realized by a change in reference value coupled with a force-parameter or cross-term change (in particular, for bond angles). Several strategies were developed to deal with these problems.

Parameter tethering is a useful methodology when the parameter values are known approximately. The method is implemented by adding harmonic terms of the form $c_i(p_i - p_i^\circ)^2$ to the merit function [eq. (1)]. Here, p_i is the current parameter value, p_i° is the assumed optimal value, and c_i is a tethering force constant. Note that the tethering should generally be removed in the final iterations to avoid unduly biasing the force field.

In the Newton–Raphson-type methods, all approximate matrix inversion techniques employed have the effect of limiting the maximum parame-

ter change. Despite this, solutions to eq. (7) sometimes yield very large parameter changes with little if any detrimental effect on the merit function. As the large steps are likely to lead to nonphysical regions of parameter space, a maximum total step (a trust radius) was implemented. The trust radius is calculated in terms of the numerical differentiation step, and was used either to scale the total step down to an acceptable value or, more commonly, to discard all solutions yielding overly large parameter changes. In the initial stages, the trust radius was set to 30 times the numerical differentiation step.

Finally, the order in which parameters were optimized was found to be important in the final results. With good reference values for bonds and angles available, these were held fixed in the initial stages. If nondiagonal terms were parameterized, these were optimized separately first, then all force constants were included, followed by a final refinement of the complete parameter set. Only in the final refinement was central differentiation utilized [eq. (8)].

Results

The automated parameterization procedure has been used to reparameterize the simple ethane molecule within MM3*, with a dual goal: first, to test the parameterization procedure and, in particular, the convergence properties resulting from the various matrix inversion techniques; and second, to estimate the quality of predictions that can be obtained in this simple system using different sources and weighting factors for the reference data. It should be noted that, for the simple ethane molecule, MM3* is identical to MM3.

The parameterization of ethane has consistently been performed using three different kinds of data: rotational barrier; structure; and Cartesian Hessian. Four different levels of quantum mechanical calculations (GAUSSIAN-94[22]) were used to obtain the reference data. Calculations were performed for the ground state and the transition state for rotation around the C—C bond (both structures in full D3d symmetry), producing fully relaxed structures, energies, and energy second derivatives at the following levels: HF/6-31G*; MP2/6-311G**; B3LYP/6-31G*; and B3LYP/6-311G**. The relatively simple Hartree–Fock calculations do not give experimental accuracy, but the behavior and deficiencies of the method are well known, and it has been widely used to produce force-field parameters.[8] Geometries at the Møller–Plesset level are expected to be very accurate[7] and may be preferred over x-ray data if crystal packing introduces significant discrepancies or hydrogen positions are needed. The hybrid HF-DFT method, B3LYP,[23] has recently been shown to give results at least comparable to MP2 at a lower computational cost,[24] and was therefore investigated here. The different sources of structure and energy data are shown in Table II. The experimental structure[25] and rotational barrier[26] were used in the parameterization of the original MM3 force field. Note that MM3 is parameterized to reproduce r_g bond lengths,[9] whereas all quantum mechanical and many force field methods yield r_e bond lengths. The differences between r_g and r_e bond lengths for the

TABLE II.
Source and Initial Data.

		Ground state				Transition state				
		Bond[a] (Å)		Angle (degrees)		Bond[a] (Å)		Angle (degrees)		ΔE[b]
Entry	Source	C—C	C—H	C—C—H	H—C—H	C—C	C—H	C—C—H	H—C—H	(kJ/mol)
1	Experimental[c]	1.534[d]	1.112[d]	111.00	107.90	—	—	—	—	12.05
2	—''—, converted[e]	1.524	1.088							
3	HF/6-31G*	1.527	1.086	111.21	107.68	1.541	1.085	111.66	107.20	12.49
4	MP2/6-311G**	1.529	1.093	111.14	107.76	1.542	1.092	111.63	107.23	12.88
5	B3LYP/6-31G*	1.531	1.096	111.35	107.53	1.545	1.095	111.81	107.04	11.79
6	B3LYP/6-311G**	1.530	1.094	111.37	107.51	1.544	1.092	111.80	107.05	11.30
7	MM3*	1.531[d]	1.113[d]	111.41	107.47	1.544	1.113	111.52	107.34	10.10

[a] Except where noted, bond lengths are r_e values. [b] Rotational barrier, potential energy difference. [c] Electron diffraction structure[25]; microwave barrier.[26] [d] r_g values. [e] Bond length conversion from r_g to r_e values calculated with MM3(92).[27]

ethane bonds have been calculated using MM3(92),[27] and have been used to generate a valid experimental comparison structure (Table II, entry 2). Note that the converted structure has not been used for parameterization, just for evaluation of the methods that yield r_e bond lengths (all QM calculated structures, and all parameter sets based on the QM structures). A comparison of Hessians and frequencies is given in Table V (in Appendix).

Two methods of combining the reference data were tested, either using calculated reference data only or combining the available experimental data with the calculated Cartesian Hessian. In the first case, the calculated structures of both the ground state and the transition state were used in the parameterization, whereas, in the second case, only the experimental gas phase structure of the ground state was utilized (cf. Table II). In both cases, the Cartesian Hessians for the calculated structures of both ground state and transition state were employed.

It is not evident *a priori* that the functional form of MM3* can reproduce the entire quantum mechanical Hessian, even for a small molecule like ethane. If not, the nondiagonal parameters employed by MM3* might simply be ''fudge'' factors in our procedure, improving the fit in a nonphysical way. All parameterizations were therefore performed first using the diagonal parameters only, using the original MM3* parameter set as initial values. The resulting force fields were tested, whereupon the nondiagonal terms were added and the parameterization continued.

Finally, the weighting factors to be used with various kinds of data were considered. The weights of different kinds of geometrical parameters were set according to the desired performance of the final force field. A discrepancy in bond length of 0.01 Å was judged to be equivalent to an error in bond angle of 0.5° or in dihedral angle of 1.0°, resulting in the weights shown in Table I. To achieve a proper balance, the energy was given a

TABLE III.
Model Systems for Ethane Parameterization.

Model	QM level	Exp. structure and barrier	Nondiagonal terms included	Hessian weight (kJ^{-1} mol $Å^2$ amu^{-1})
1	HF / 6-31G*	no	no	0.01
2	HF / 6-31G*	yes	no	0.01
3	HF / 6-31G*	no	no	0.1
4	HF / 6-31G*	yes	no	0.1
5	HF / 6-31G*	no	no	1.0
6	HF / 6-31G*	yes	no	1.0
7	MP2 / 6-311G**	no	no	0.01
8	MP2 / 6-311G**	yes	no	0.01
9	B3LYP / 6-31G*	no	no	0.01
10	B3LYP / 6-31G*	yes	no	0.01
11	B3LYP / 6-311G**	no	no	0.01
12	B3LYP / 6-311G**	yes	no	0.01
13	HF / 6-31G*	no	yes	0.01
14	HF / 6-31G*	yes	yes	0.01
15	HF / 6-31G*	no	yes	0.1
16	HF / 6-31G*	yes	yes	0.1
17	HF / 6-31G*	no	yes	1.0
18	HF / 6-31G*	yes	yes	1.0
19	MP2 / 6-311G**	no	yes	0.01
20	MP2 / 6-311G**	yes	yes	0.01
21	B3LYP / 6-31G*	no	yes	0.01
22	B3LYP / 6-31G*	yes	yes	0.01
23	B3LYP / 6-311G**	no	yes	0.01
24	B3LYP / 6-311G**	yes	yes	0.01

very high scale factor, to offset the fact that only one potential energy difference (the rotational barrier) was used in the parameterization. Finally, the energy second derivatives (the Hessian data) had to be balanced with respect to the remaining data types. Hagler et al.[8] used a factor of 15,000 between weights for energies and Hessian. We obtained our basic factor for Hessian data by dividing the energy scale factor by 10,000, but also tested two higher weights for the HF/6-31G* Hessian. In total, we tested 24 different parameterization systems (Table III).

The model systems were evaluated both by how closely the reference data could be reproduced, and by how well the final force field reproduced the experimental vibrational spectrum of ethane. The results are summarized in Table IV. The final parameter sets for all model systems are shown in Tables VI and VII (in Appendix).

It can be seen from Table IV that, in many cases, an almost perfect fit could be obtained to the reference data. The weighting factor for the rotational barrier may have been unduly high, as in all cases the rotational barrier used as reference was reproduced exactly. On the other hand, it can be seen that a higher weighting factor for the Hessian elements had a detrimental effect on calculated structures (models 3–6 and 15–18). This effect was also noted by Hagler et al.[8] Interestingly, overweighting of the Hessian also has a detrimental effect on the calculated frequencies, particularly when nondiagonal terms are included in the parameterization (models 17 and 18). Use of an intermediate Hessian weighting factor improves the

TABLE IV.
Summary of Final Results, Structural and Vibrational Data.

Model	Compared to (Table II entry):	Bond[a] (rms, Å)	Angle[a] (rms, degrees)	ΔE (error, kJ / mol)	Hessian[b] Slope	Hessian[b] r^2	Frequencies[c] Slope	Frequencies[c] r^2
1	3	0.001	0.2	0.0	1.00	99.7%	0.906	99.50%
2	1	0.001	0.0	0.0	1.00	99.6%	0.963	98.42%
3	3	0.031	1.6	0.0	1.00	99.7%	0.899	99.71%
4	1	0.046	2.4	0.0	1.00	99.7%	0.907	99.77%
5	3	0.054	6.7	0.0	1.00	99.8%	0.884	98.92%
6	1	0.061	6.6	0.0	1.00	99.8%	0.883	98.90%
7	4	0.001	0.2	0.0	1.00	99.7%	0.946	99.27%
8	1	0.001	0.0	0.0	1.00	99.7%	0.991	99.55%
9	5	0.001	0.2	0.0	1.00	99.7%	0.957	99.48%
10	1	0.001	0.0	0.0	1.00	99.7%	0.995	99.42%
11	6	0.001	0.2	0.0	1.00	99.6%	0.966	99.44%
12	1	0.001	0.0	0.0	1.00	99.6%	1.011	99.40%
13	3	0.001	0.2	0.0	1.00	99.7%	0.906	99.38%
14	1	0.001	0.0	0.0	1.00	99.7%	0.962	98.09%
15	3	0.003	0.7	0.0	1.00	99.7%	0.902	99.46%
16	1	0.026	1.3	0.0	1.00	99.7%	0.911	99.64%
17	3	0.067	9.7	0.0	1.00	99.8%	0.877	96.39%
18	1	0.068	9.6	0.0	1.00	99.8%	0.872	96.16%
19	4	0.001	0.2	0.0	1.00	99.7%	0.947	99.16%
20	1	0.001	0.0	0.0	1.00	99.7%	0.990	99.33%
21	5	0.001	0.1	0.0	1.00	99.7%	0.957	99.24%
22	1	0.001	0.0	0.0	1.00	99.7%	0.994	99.21%
23	6	0.001	0.1	0.0	1.00	99.7%	0.967	99.19%
24	1	0.001	0.0	0.0	1.00	99.7%	1.011	99.16%

[a] Only ground state structure included when compared with experimental data ("1" in column 2). [b] Regression between Cartesian energy second derivatives from MM3* and the corresponding QM source, no intercept used. [c] Regression between experimental and calculated frequencies, no intercept used. Force field frequencies were also calculated in MM3(92)[27] to assign symmetries. Numerical frequencies in MM3* and analytical frequencies in MM3(92) agree within 2 cm^{-1}.

calculated frequencies, but with an unacceptable loss of structural accuracy (models 3 and 4 and 15 and 16).

Looking at the cases where only diagonal terms were parameterized, with a low weighting of the Hessian elements (models 1 and 2 and 7–12), we see that all input data are very closely reproduced. The high correlation for the Hessians is gratifying, as no van der Waals parameters have been modified. Discrepancies in the van der Waals or electrostatic parameters would be expected to give Hessian errors that could not be compensated by linear combinations of other terms. In the case where parameterization is based entirely upon calculated data (models 1, 7, 9, and 11), the frequencies show approximately the same slope as the original QM results. However, when the QM Hessian is combined with experimental structures and energy (models 2, 8, 10, and 12), *without including any experimental frequency information*, the calculated frequencies are much improved. The correlation between between experimental and calculated frequencies stays approximately the same, but the slope in the regression now approaches unity.

The inclusion of nondiagonal terms in the parameterization does not seem to improve the results in any way. The main reason to include them would be to improve the calculated frequencies, but in all cases the correlation with experimental frequencies actually decreases when they are allowed to vary. Therefore, it seems that the set of cross-terms employed by MM3 is not optimal for describing the quantum mechanical potential energy surfaces. It is noteworthy that even the correlation between the force field and corresopnding quantum-mechanically calculated frequencies decreases when nondiagonal terms are included in the parameterization. The conclusion must be that the MM3* nondiagonal terms are used in the parameterization to describe features of the Hessian that should properly be described by other terms, which are unknown. On the other hand, the MM3 functional form, when properly employed, does very well in describing experimental frequencies (Table V, see Appendix). The recommendation must therefore be to modify the nondiagonal terms only when more appropriate data can be reliably included in the parameterization.

From the limited study performed here, the B3LYP/6-31G* method seems to have the best cost/performance ratio of the investigated methods for generating parameterization data, as it is substantially cheaper than both B3LYP/6-311G** and MP2/6-311G**, and gives a more accurate force field than does HF/6-31G* at a very similar cost. In particular, when used in conjunction with experimental data for parameterization of diagonal terms only (model 10), the fit with experiment is excellent.

It must be pointed out that the parameters derived here are *not* meant to supplant the current MM3 parameters. In the current study, the parameters have been derived for one molecule only, whereas the original MM3 parameters were designed to fit a large range of systems. The current results should only be seen as a survey of methods for parameterization. Only for the rotational barrier of ethane do the parameter sets derived here substantially improve the original MM3 results and, for calculated frequencies, the results are worse.

PROCEDURE EVALUATION

It was noted that it was very beneficial to tailor the Newton–Raphson numerical differentiation step for each parameter (also used for the simplex initialization). Best performance was obtained when all parameter steps yielded merit function changes within one or two orders of magnitude of each other. If one or a few parameters were given overly large steps, the Newton–Raphson method was very slow to converge, whereas the simplex method eventually self-corrected by contracting in the offending search direction. The weighted simplex method compared very well with the numerical Newton–Raphson method in the initial stages. For final convergence, and for verification that a real optimum had been reached, the parameterization was always finalized with Newton–Raphson optimization using central differences.

The different schemes for generating the Newton–Raphson step were evaluated statistically, by comparing how many times each method yielded the best step. Somewhat surprisingly, SVD was found to have the worst performance. The addition of Lagrange multipliers yielded the best result 218 times, compared with 186 times for the Levenberg–Marquardt method and only 58 times for SVD. However, in almost all cases, all three methods yielded improvements to the merit function. The final optima would have been found with any of the methods, probably with little difference in actual parameterization time.

One caveat must be mentioned for the Levenberg–Marquardt method; if one of the parameter changes fails to affect the merit function (which could happen, e.g., for a bond length if the corre-

sponding force constant had been set to zero in the previous step), the matrix **A** becomes singular and stays singular, irrespective of diagonal multiplication. In this case, the Levenberg-Marquardt method fails to find *any* solution. Thus, the most general method would seem to be the addition of Lagrange multipliers [eq. (9)]. It should be noted that this method is well suited for a one-dimensional automatic optimization of the Lagrange multiplier in each iteration. This method is currently being implemented, but has not yet been evaluated.

Conclusions

In the current work, a very efficient concept for automatic parameterization of molecular mechanics force fields has been described. Any type of reference data that can be predicted by the force field can, in principle, be used in the parameterization. The entire set of procedures, together with a sample application, has been made generally available.

The recommended method for parameterization is to use available experimental data (both structure and energy data) together with Cartesian energy second derivatives from calculations at B3LYP/6-31G* or a similar level of theory. This combination seems to give good performance at a reasonably low cost. For large parameter sets, the recommended strategy is to optimize subsets of no more than 30–40 parameters with the simplex method followed by a Newton–Raphson method using forward differences. When subsets are optimized, the full set of parameters should be subjected to a Newton–Raphson optimization using central differences, possibly preceded by a few iterations using forward differences. At all stages, the numerical differentiation steps should be tailored to give similar merit function changes (within a few orders of magnitude).

The methods described herein have been tested in widely different fields. These include parameter sets for new heterocyclic systems and various organometallic complexes, and for transition states. The special problems encountered in each of these fields will be discussed in separate publications.

Acknowledgments

The authors thank Helena Hagelin, Peter Korall, Christian Linde, and Wesley Schaal for testing the automation procedures described herein. Thanks are also due to Peter Shenkin for much help with MacroModel.

Appendix

IMPLEMENTATION DETAILS

The full set of procedures defining the parameterization method, together with an example application, are available from the authors.[29] The central item in the parameterization is the results matrix, **R**, defined in eq. (10):

$$\mathbf{R} \equiv \begin{pmatrix} w_1 & y_1^\circ & \hat{y}_1(\bar{\mathbf{p}}_0) & \hat{y}_1(\bar{\mathbf{p}}_1) & \cdots \\ w_2 & y_2^\circ & \hat{y}_2(\bar{\mathbf{p}}_0) & \hat{y}_2(\bar{\mathbf{p}}_1) & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix} \quad (10)$$

The matrix **R** consists of one row for each data point to be used in the parameterization, with data from different parameter sets in tab-delimited columns. The first column contains weighting factors, and the second column consists of reference data points to be reproduced as closely as possible by the parameter set. The remaining columns contain calculated values for all data points with different parameter sets. The results matrix is created from data from BatchMin executions by several Unix scripts (*vide infra*). The data points are generated for one set of parameters at a time, producing a single column that is appended to **R** with the Unix ''paste'' command. The parameter evaluation routine reads the matrix **R** and accumulates weighted sums of squares of deviations from the reference values [χ_j^2, eq. (11)]. If numerical differentiation has been requested, the routine also accumulates the corresponding weighted squared deviations between columns of calculated data points.

$$\chi^2(\bar{\mathbf{p}}_j) = \sum_i \left(w_i \left[y_i^\circ - \hat{y}_i(\bar{\mathbf{p}}_j) \right] \right)^2 \quad (11)$$

For dihedral angle data points, eq. (11) must be modified, as the observables are cyclic; that is, a calculated absolute deviation larger than 180° should shifted by 360°. This shift is implemented for data points with a negative scale factor w_i (only the absolute value of the scale factor is used for scaling, the sign is only used to signal the special case of dihedral angle data points to the evaluation routine).

The definition of a parameter subset to be refined is the parameter listing matrix, **P**, defined in eq. (12):

$$\mathbf{P} \equiv \begin{pmatrix} row_1 & position_1 & p_1 & [diff_1] \\ row_2 & position_2 & p_2 & [diff_2] \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix} \quad (12)$$

The matrix **P** defines the parameters to be modified by row and position in the MacroModel force field file. This way of identifying parameters in the force field is simple and efficient, but quite prone to error when the force field is modified in such a way that the number of lines changes. It is strongly recommended that the consistency between force-field file and parameter lists is checked after any manual force-field change.

The value of the parameters are given in column 3 of matrix **P**. Column 2 defines which position on the force field line to use for a parameter, but is also used to identify parameters that may receive a negative value. Many parameters, like force constants and reference bond lengths, should generally not be negative, and the parameter modification procedure will accordingly refuse to handle negative parameters unless the position number in the parameter listing file is negative. The optional constant in column 4 is used as the numerical differentiation step, if present.

STRUCTURAL DATA

A complete listing of all structural elements can be generated by BatchMin using the ELST command. We identify the relevant structural elements by adding a unique text sequence in the comment lines of the force field file. The comment is copied to the structure listing file, with one line for each structural element and its associated comment. The desired data lines can then be extracted from the output file with the Unix ''grep'' facility (identifying the lines with the unique text sequence) and the actual value separated with Unix ''awk.'' For this simple procedure to succeed, it is very important that the listing is consistent between runs, with the same parameters always coming in the same order. Some BatchMin commands (like the MINI command, which performs the geometry optimization) will affect which structural elements are actually output (e.g., torsional angles with all parameters set to zero are deleted from the listing). Thus, to get the same type of listing for the reference and optimized structures, it is necessary to issue an otherwise unneeded ''MINI'' command with zero iterations before listing the reference structural elements. The command sections we use for listing of structural elements is shown in Figure 1.

Note that if a parameter (e.g., for a torsion) is set exactly to zero, the corresponding line will not be output and the order of the list will change. We therefore ensure that at least one parameter for each structural element of interest is always nonzero; for example, by setting a v_1 term that is not being refined to a negligible value larger than zero.

When calculating structural elements from the reference structures, a Unix script will identify bond lengths, bond angles, and dihedral angles from the format of the BatchMin output file and initialize the results matrix **R** with the appropriate weighting factor in the first column (negative for dihedral angles, *vide supra*).

ENERGY DATA

Relative energies between conformations or between ground state and transition state are calculated from a representative conformational ensemble using an approximate Boltzmann distribution determined from steric energies [eq. (1)]. Each data point is calculated from one MacroModel structure file containing all representatives of the two ensembles to be compared. Each structure in the file is named, with the first letter of the name determining to which ensemble the structure belongs (note that to use more than two different initial

FFLD	2	0	0	0	0.0000	0.0000	0.0000	0.0000
READ								
MINI	9	0	xx[a]	0	0.0000	0.0000	0.0000	0.0000
ELST	1	0	0	0	0.0000	0.0000	0.0000	0.0000

FIGURE 1. Lines 3–6 of the command files used to generate structure element listings. [a] For optimization prior to listing, set xx to 50. For listing of structural elements from a reference structure, set xx to 0.

```
FFLD      2      0      0      0    0.0000    0.0000    0.0000    0.0000
BGIN
READ
MINI      9      0     50      0    0.0000    0.0000    0.0000    0.0000
END
```

FIGURE 2. Lines 3–7 of the command file used to generate conformational energies.

letters in structure names within one file will give erroneous results). The command file utilized for energetic calculations is shown in Figure 2.

The conformational energies are extracted from the BatchMin log file together with structure names and the result stored in a temporary file, which in turn is read by a C-program calculating the Boltzmann distributions and the corresponding relative energies.

Usually, only a few relative energies are included in the parameterization, whereas the number of structural elements and/or energy derivatives is orders of magnitude larger. To offset the resulting bias, the energies are given a fairly high scale factor (Table I).

ENERGY DERIVATIVES

The obvious choice for calculating the energy derivatives is the DLST command, which generates a complete list of numerically and analytically calculated first and second Cartesian energy derivatives. However, this command cannot be used, because it is based on a modified paradigm! Internally, BatchMin stores hydrogens not at the nuclear position but at the van der Waals offset positions.[30] The energy derivatives output by the DLST command are calculated for the offset positions, and not for the Cartesian coordinates used in input and output files (the latter positions are used both by original MM3 and by all quantum mechanical calculations). Thus, second derivatives involving at least one hydrogen will deviate by a factor of between 1 and 1.2 from the values based on nuclear positions, depending on which interaction types are dominant. The ''correct'' second derivatives are calculated numerically for use with the frequency calculation routine, and can be output in mass-weighted form using a combination of DEBG and RRHO commands (Fig. 3).

The MINI command performs the actual calculation of the energy derivatives to be output by RRHO. The two DEBG commands are needed first to enable RRHO operation on nonoptimized structures and second to force verbose RRHO output. The values are extracted from the output using the Unix ''awk'' command and transferred to the results file in lower triangular form. A C-program is available for automatic extraction and mass weighting of the corresponding second derivatives in a formatted GAUSSIAN checkpoint file.

EVALUATION OF RESULTS FILE

The evaluation routine is implemented as a C-program with several options. The basic mode of operation is to read the results file one line at a time and accumulate the merit function for each force field [eq. (2)]. This mode is used for evaluations of trial solutions, and also to generate function values to the Simplex script.

When numerical differentation is used, the third column of matrix **R** always contains the results from the current force field, whereas the remaining columns are calculated from parameter sets where one parameter has been shifted slightly (as a de-

```
FFLD      2      0      0      0    0.0000    0.0000    0.0000    0.0000
READ
DEBG    210
DEBG    211
MINI      9      0      0      0    0.0000    0.0000    0.0000    0.0000
RRHO      0      0      0      0    0.0000    0.0000    0.0000    0.0000
```

FIGURE 3. Lines 3–8 of the command file used to generate mass-weighted energy derivatives.

fault, the shift is 1% of the current value, but not less than 0.01). For forward differences, each parameter is increased in order [eq. (13)], whereas, for central differences, one parameter is first increased, then decreased by the same value before continuing with the next parameter [eq. (14)]:

$$\mathbf{R} \equiv \begin{pmatrix} w_1 & y_1^\circ & \hat{y}_1(\bar{\mathbf{p}}) & \hat{y}_1(\bar{\mathbf{p}} + \delta p_1) & \hat{y}_1(\bar{\mathbf{p}} + \delta p_2) & \cdots \\ w_2 & y_2^\circ & \hat{y}_2(\bar{\mathbf{p}}) & \hat{y}_2(\bar{\mathbf{p}} + \delta p_1) & \hat{y}_2(\bar{\mathbf{p}} + \delta p_2) & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \quad \text{forward differences} \tag{13}$$

$$\mathbf{R} \equiv \begin{pmatrix} w_1 & y_1^\circ & \hat{y}_1(\bar{\mathbf{p}}) & \hat{y}_1(\bar{\mathbf{p}} + \delta p_1) & \hat{y}_1(\bar{\mathbf{p}} - \delta p_1) & \hat{y}_1(\bar{\mathbf{p}} + \delta p_2) & \cdots \\ w_2 & y_2^\circ & \hat{y}_2(\bar{\mathbf{p}}) & \hat{y}_2(\bar{\mathbf{p}} + \delta p_1) & \hat{y}_2(\bar{\mathbf{p}} - \delta p_1) & \hat{y}_2(\bar{\mathbf{p}} + \delta p_2) & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}, \quad \text{central differences} \tag{14}$$

In the differentiation and subsequent usage of the derivatives, the shift used for each parameter will be defined as a unit step for that parameter. Thus, the shift actually used must be saved and used to scale the calculated step when determining new trial parameter sets. When in this mode, the evaluation routine will not only accumulate the merit functions, but will also calculate approximate derivatives according to eq. (15). The calculated derivatives are not saved as such, but directly accumulated in the matrix $\mathbf{A}$ [eq. (16)] as well as in the vector $\bar{\mathbf{b}} \equiv \mathbf{J}^T\bar{\mathbf{r}}$ [eq. (17)]:

$$\frac{\partial \hat{y}_i}{\partial p_j} = \hat{y}_i(\bar{\mathbf{p}} + \delta p_j) - \hat{y}_i(\bar{\mathbf{p}})$$

(forward differentiation)

$$\frac{\partial \hat{y}_i}{\partial p_j} = \frac{1}{2}\left[\hat{y}_i(\bar{\mathbf{p}} + \delta p_j) - \hat{y}_i(\bar{\mathbf{p}} - \delta p_j)\right]$$

(central differnetiation)

(15)

$$A_{jk} = \sum_i w_i^2 \frac{\partial \hat{y}_i}{\partial p_j}\frac{\partial \hat{y}_i}{\partial p_k} \tag{16}$$

$$b_j = \sum_i w_i^2 \frac{\partial \hat{y}_i}{\partial p_j}(y_i^\circ - \hat{y}_i) \tag{17}$$

When used in central differentiation mode, the final merit function values can also be used to calculate absolute derivatives according to eqs. (18) and (19) (note that δp_j is defined as a unit step for parameter p_j):

$$\frac{d\chi^2}{dp_j} = \frac{1}{2}\left[\chi^2(\bar{\mathbf{p}} + \delta p_j) - \chi^2(\bar{\mathbf{p}} - \delta p_j)\right] \tag{18}$$

$$\frac{d^2\chi^2}{dp_j^2} = \chi^2(\bar{\mathbf{p}} + \delta p_j) + \chi^2(\bar{\mathbf{p}} - \delta p_j) - 2\chi^2(\bar{\mathbf{p}}) \tag{19}$$

It is now possible to calculate a one-dimensional Newton–Raphson step for each parameter [eq. (20)]:

$$\Delta p_j = -\frac{d\chi^2/dp_j}{d^2\chi^2/dp_j^2} \tag{20}$$

Note that, to actually calculate new parameter values, it is also necessary to scale by the original parameter shift [eq. (21)]:

$$p_{j,\,new} = p_{j,\,old} + \Delta p_j\,\delta p_j \tag{21}$$

Eq. (20) is not very efficient for generation of new parameter sets, but can be used as one of the convergence criteria. At a true optimum, for each parameter, the merit function second derivative [eq. (19)] should be positive, and the step calculated in eq. (20) should be small (less than 1; that is, less than the shift value used in the numerical differentiation). Note that these tests are one dimensional. Full convergence is only reached when, in addition to the above tests, the multidimensional methods fail to improve the merit function significantly.

With the accumulated matrix $\mathbf{A}$ available, the evaluation routine will create several modified matrix inverses $(\mathbf{A}')^{-1}$, either directly by SVD or by diagonal modification followed (because $\mathbf{A}'$ is symmetric and positive definite) by Cholesky decomposition.[20] The evaluation routine will output a file where each row contains the parameter steps $\bar{\mathbf{x}}$ calculated from eq. (22) for one trial set.

$$\bar{\mathbf{x}} = (\mathbf{A}')^{-1}\bar{\mathbf{b}} \tag{22}$$

FORCE FIELD MODIFICATION

The parameter modification routine is responsible for transferring data between the parameter listing file [eq. (12)] and the force field file, and also for modifying the parameter lists. The initial parameter list, identifying which parameters in the force field to refine and also for which parameters to allow negative values, must be created manually. Using one parameter list file as a template, the parameter modification routine can output a new parameter list containing the parameter values from an arbitrary input force field file, or transfer the data from a parameter listing to the appropriate positions in the force field file.

The first step in one parameterization iteration consists of using the parameter modification routine to generate a parameter listing where each parameter has been increased by a small value suitable for numerical differentiation. The resulting file can then be used to create the required force fields where only one parameter has been modified. After evaluation of all force fields, the parameter modification routine will read the evaluation output, the initial and the shifted parameter lists, and generate several new parameter listings using eq. (21). Each new parameter set is then evaluated. If no improvement is found, the procedure quits, otherwise it starts a new cycle using the best parameter set so far.

SIMPLEX IMPLEMENTATION

The simplex procedure is initialized by generating $N + 1$ independent parameter sets for N variable parameters. In the current implementation, the procedure for forward numerical differentiation is used, so that each apex corresponds to a shift in one parameter only. This will give rise to a right-angled simplex in parameter space. Each apex is represented by one column in a results matrix **R** [eq. (10)], with a corresponding parameter listing file. As the simplex is moved through parameter space, columns in **R** will be substituted using the Unix ''cut'' and ''paste'' facilities.

The first step in moving the simplex consists of evaluating the merit function at each apex, and identifying the worst parameter set ($\bar{\mathbf{p}}_W$). The second worst ($\bar{\mathbf{p}}_N$) and best ($\bar{\mathbf{p}}_B$) parameter sets should also be identified for later comparisons. An inversion point ($\bar{\mathbf{p}}_I$) is generated by taking the average of all parameter sets except $\bar{\mathbf{p}}_W$ [eq. (23)], whereupon a new parameter set $\bar{\mathbf{p}}_R$ is calculated by reflecting $\bar{\mathbf{p}}_W$ through $\bar{\mathbf{p}}_I$ [eq. (24)]:

$$\bar{\mathbf{p}}_I = \frac{1}{N} \sum_{j \neq W} \bar{\mathbf{p}}_j \tag{23}$$

$$\bar{\mathbf{p}}_R = 2\bar{\mathbf{p}}_I - \bar{\mathbf{p}}_W \tag{24}$$

The subsequent scheme depends on the merit function value at the new point, according to the options in Figure 4.

The entire procedure is terminated when no improvement to $\bar{\mathbf{p}}_B$ has been obtained for $3N$ cycles (or another convenient number of cycles). The Simplex may not be able to find better values due to excessive contractions in one or a few dimensions. The termination of a simplex should therefore not be taken as a sign that an optimum parameter set has been reached. The convergence could be tested by restarting a full simplex from the last $\bar{\mathbf{p}}_B$, or by restoring to the full central differentiation scheme (*vide supra*).

The locally developed weighted simplex scheme is identical to the standard Simplex in all respects except the definition of the inversion point $\bar{\mathbf{p}}_I$. One of the most important features of the standard simplex method is that the hypervolume of the simplex is kept constant by a reflection, and halved

1) If $\bar{\mathbf{p}}_R$ is better than $\bar{\mathbf{p}}_B$, attempt an expansion. Calculate $\bar{\mathbf{p}}_E = 3\bar{\mathbf{p}}_I - 2\bar{\mathbf{p}}_W$. Replace $\bar{\mathbf{p}}_W$ by the best set of $\bar{\mathbf{p}}_E$ and $\bar{\mathbf{p}}_R$, continue with the next cycle.

2) If $\bar{\mathbf{p}}_R$ is better than $\bar{\mathbf{p}}_N$, the simple reflection succeeded. Replace $\bar{\mathbf{p}}_W$ by $\bar{\mathbf{p}}_R$, continue with the next cycle.

3) Calculate contraction $\bar{\mathbf{p}}_K$, by one of two possible formulas. If $\bar{\mathbf{p}}_R$ is worse than $\bar{\mathbf{p}}_W$, set $\bar{\mathbf{p}}_K = (\bar{\mathbf{p}}_I + \bar{\mathbf{p}}_W)/2$, else set $\bar{\mathbf{p}}_K = (3\bar{\mathbf{p}}_I - \bar{\mathbf{p}}_W)/2$. If $\bar{\mathbf{p}}_K$ is better than $\bar{\mathbf{p}}_N$, replace $\bar{\mathbf{p}}_W$ by $\bar{\mathbf{p}}_K$, continue with the next cycle. If not, do a massive contraction. Replace each parameter set $\bar{\mathbf{p}}_j$ by $(\bar{\mathbf{p}}_j + \bar{\mathbf{p}}_B)/2$, restart with the new contracted simplex.

FIGURE 4. Simplex operation after reflection, use lowest valid option only.

TABLE V.
Comparison of Sources of Parameterization Data.

Model	Bond (rms, Å)	Angle (rms, degree)	ΔE (error, kJ / mol)	Hessian[a]		Frequency[b]	
				Slope	r^2	Slope	r^2
MM3*	0.002[c]	0.4	−2.0	—	—	1.008	99.74%
HF / 6-31G*	0.003[d]	0.2	0.4	0.97	99.2%	0.906	99.77%
MP2 / 6-311G**	0.005[d]	0.1	0.8	0.99	99.5%	0.946	99.71%
B3LYP / 6-31G*	0.007[d]	0.4	−0.3	1.01	99.5%	0.957	99.79%
B3LYP / 6-311G**	0.006[d]	0.4	−0.7	1.04	99.5%	0.967	99.79%

[a] Regression between Cartesian energy second derivatives from MM3* and the corresponding QM source, no intercept used. [b] Regression vs. experimental frequencies, no intercept used. MM3(92) was used in lieu of MM3* for frequency calculations to allow assignment of symmetries. [c] Comparison to r_g values (Table II, entry 1). [d] Comparison to r_e values (Table II, entry 2).

or doubled, respectively, by a contraction or expansion. Due to this hypervolume consistency, linear dependencies between the parameter sets are avoided. However, the inversion point defined in eq. (23) is not the only point with this property. Any point in the hyperplane obtained by excluding $\bar{\mathbf{p}}_W$ from the average would be equally valid. The value of the merit function has therefore been used to bias the inversion point toward the best known points, [eq. (25)]:

$$\bar{\mathbf{p}}_I = \frac{\sum \bar{\mathbf{p}}_j \left[\chi^2(\bar{\mathbf{p}}_j) - \chi^2(\bar{\mathbf{p}}_W) \right]}{\sum \left[\chi^2(\bar{\mathbf{p}}_j) - \chi^2(\bar{\mathbf{p}}_W) \right]} \qquad (25)$$

TABLE VI.
Optimized Diagonal Parameters.

	Bond				Angle				Torsion
	C—C		C—H		H—C—H		C—C—H		H—C—C—H
Model	l_0	k	l_0	k	θ_0	k	θ_0	k	v_3
MM3*	1.5247	4.4900	1.1120	4.7400	107.8000	0.5500	109.3100	0.5900	0.2380
1	1.5222	4.1246	1.0842	5.8434	105.9892	0.7347	108.4986	0.7603	0.2988
2	1.524	3.8355	1.1091	4.8505	108.3622	0.7226	110.5974	0.7876	0.2899
3	1.4787	5.6184	1.0809	5.9809	107.4864	0.6833	106.7712	0.8092	0.2758
4	1.4678	6.1151	1.0843	5.8215	108.0663	0.68	105.2066	0.8949	0.2468
5	1.4421	7.6434	1.075	6.2336	107.9169	0.6735	99.2773	1.1822	0.1671
6	1.4448	7.5115	1.0748	6.2428	107.9239	0.6731	99.1608	1.1898	0.1544
7	1.5235	3.922	1.0922	5.4661	106.1182	0.631	108.3024	0.6528	0.3088
8	1.5268	3.6015	1.1099	4.7838	107.2655	0.6455	109.1933	0.6994	0.2882
9	1.5256	3.6665	1.0949	5.2808	105.7387	0.6547	108.4272	0.6839	0.2817
10	1.527	3.3819	1.1101	4.7096	107.1385	0.6533	109.1244	0.7226	0.2889
11	1.5238	3.5918	1.0921	5.2052	106.1089	0.6134	108.8434	0.6467	0.2698
12	1.5255	3.2557	1.1099	4.5509	107.4642	0.6208	109.4731	0.7008	0.2901
13	1.5154	3.8147	1.0835	5.8402	107.007	0.7056	109.369	0.7374	0.3003
14	1.5243	3.5913	1.1093	4.8191	109.5268	0.6929	112.3319	0.7003	0.291
15	1.5054	4.1878	1.0802	5.9929	97.1611	1.0478	92.7156	1.3862	0.3157
16	1.497	4.5999	1.0852	4.7673	107.1618	0.6978	104.9739	0.808	0.4189
17	1.4251	10.2573	1.0825	5.8913	126.7472	0.2931	99.1774	1.357	−0.0149
18	1.4305	10.0568	1.0838	5.8338	113.0908	0.3151	98.0123	1.3252	−0.0145
19	1.5143	3.6134	1.091	5.4845	105.8159	0.6455	107.7351	0.7031	0.3104
20	1.5221	3.4363	1.1094	4.7762	107.5019	0.643	109.389	0.7328	0.2885
21	1.5156	3.3069	1.0939	5.31	106.0657	0.6459	108.6587	0.7004	0.2845
22	1.5213	3.1924	1.1096	4.7164	107.9642	0.634	109.8498	0.7064	0.289
23	1.5159	3.1472	1.0913	5.2238	106.3099	0.6086	108.9558	0.6751	0.2689
24	1.521	3.0555	1.1096	4.5525	108.0802	0.606	110.0164	0.7117	0.2925

Note that $\bar{\mathbf{p}}_W$ no longer needs to be excluded from the summation, as it will always get a coefficient of zero. We have found the weighted simplex to orient more rapidly than the standard simplex in the direction of greatest improvement. For modest parameterization with up to ca. 40 parameters, the performance is at least equal to the numerical Newton–Raphson methods discussed previously, except possibly in the last stages where the merit function is close to a quadratic form.

ETHANE PARAMETERIZATION DETAILS

Table V shows a summary of the quality of the various sources of parameterization data. As can be seen, all methods calculate the structure with close to experimental accuracy. The high-level QM methods seem to give bond lengths that are slightly too long, but it should be remembered that the comparison is to r_e values that have been calculated from the experimental r_g values using the MM3(92) anharmonicity. With the added uncertainty in this conversion, all bond length values must be considered to be within experimental accuracy.

All quantum mechanical Hessians are compared with the MM3* Hessian, as MM3* gives the closest correspondence with experimental frequencies. It can be seen from the correlation that all Hessians agree on the shape of the PES around the stationary points, but differ slightly in magnitude. As expected, the HF Hessian gives overly high force constants, and should be scaled in order to achieve the best fit with experiment. From the correlation with experimental frequencies, it can be observed that all QM methods would benefit from a similar scaling, even though, as observed previously, the scale factors for the correlated methods should be closer to unity.[28] Here, we have used the calculated Hessians without scaling, and evaluate the resulting force field by the fit to experimental data, not the absolute deviations in calculated frequencies.

TABLE VII.
Optimized Nondiagonal Parameters.

	Angle			Torsion
	H—C—H	C—C—H		H—C—C—H
Model	k_{bb}	k_{bb}	k_{sb}	k_{ts}
MM3*	0.0000	0.3000	0.0800	0.0590
13	0.0012	0.4066	−0.0314	0.0396
14	0.0605	0.5291	−0.0166	0.057
15	0.3421	0.7627	−0.0155	0.0831
16	0.1496	0.5769	0.0049	0.2964
17	−0.4884	−0.0423	0.2218	−0.059
18	−0.8042	−0.3769	0.2343	−0.0415
19	−0.0448	0.3405	−0.0353	0.0353
20	−0.0746	0.3355	−0.0213	0.0431
21	−0.0754	0.313	−0.0334	0.0386
22	−0.0396	0.3837	−0.0163	0.043
23	−0.0942	0.2969	−0.0311	0.0294
24	−0.0942	0.3252	−0.0164	0.0512

References

1. U. Burkert and N. L. Allinger, *Molecular Mechanics*, ACS Monograph 177, ACS, Washington, DC, 1982.
2. J. P. Bowen and N. L. Allinger, In *Reviews in Computational Chemistry, Vol. 9*, K. B. Lipkowitz and D. B. Boyd, Eds., VCH, New York, 1991, p. 81.
3. A. Warshel and S. Lifson, *J. Chem. Phys.*, **49**, 5116 (1968); A. Warshel and S. Lifson, *J. Chem. Phys.*, **53**, 582 (1970).
4. J. L. M. Dillen, *J. Comput. Chem.*, **11**, 1125 (1990); J. L. M. Dillen, *J. Comput. Chem.*, **13**, 257 (1992).
5. S. B. Engelsen, J. Fabricius, and K. Rasmussen, *Acta Chem. Scand.*, **48**, 548 (1994). S. B. Engelsen, J. Fabricius, and K. Rasmussen, *Acta Chem. Scand.*, **48**, 553 (1994).
6. K. Palmö, L.-O. Pietilä, and S. Krimm, *Comput. Chem.*, **15**, 249 (1991).
7. T. A. Halgren, *J. Am. Chem. Soc.*, **114**, 7827 (1992); T. A. Halgren, *J. Comput. Chem.*, **17**, 490 (1996); T. A. Halgren, *J. Comput. Chem.*, **17**, 520 (1996); T. A. Halgren, *J. Comput. Chem.*, **17**, 553 (1996); T. A. Halgren and R. N. Nachbar, *J. Comput. Chem.*, **17**, 587 (1996); T. A. Halgren, *J. Comput. Chem.*, **17**, 616 (1996).
8. J. R. Maple, M.-J. Hwang, T. P. Stockfisch, U. Dinur, M. Waldman, C. S. Ewig, and A. T. Hagler, *J. Comput. Chem.*, **15**, 162 (1994); M.-J. Hwang, T. P. Stockfisch, and A. T. Hagler, *J. Am. Chem. Soc.*, **116**, 2515 (1994).
9. N. L. Allinger, *J. Am. Chem. Soc.*, **99**, 8127 (1977); N. L. Allinger, H. J. Geise, W. Pyckhout, L. A. Paquette, and J. C. Gallucci, *J. Am. Chem. Soc.*, **111**, 1106 (1989); N. L. Allinger, Y. H. Yuh, and J.-H. Lii, *J. Am. Chem. Soc.*, **111**, 8551 (1989); J.-H. Lii and N. L. Allinger, *J. Am. Chem. Soc.*, **111**, 8556 (1989); J.-H. Lii and N. L. Allinger, *J. Am. Chem. Soc.*, **111**, 8576 (1989); N. L. Allinger, K. Chen, and J.-H. Lii, *J. Comput. Chem.*, **17**, 642 (1996). N. L. Allinger, K. Chen, and J.-H. Lii, *J. Comput. Chem.*, **17**, 642 (1996); N. Nevins, K. Chen, and N. L. Allinger, *J. Comput. Chem.*, **17**, 669 (1996); N. Nevins, J.-H. Lii, and N. L. Allinger, *J. Comput. Chem.*, **17**, 695 (1996); N. Nevins and N. L. Allinger, *J. Comput. Chem.*, **17**, 730 (1996); N. L. Allinger, K. Chen, J. A. Katzenellenbogen, S. R. Wilson, and G. M. Anstead, *J. Comput. Chem.*, **17**, 747 (1996); G. Liang, P. C. Fox, and J. P. Bowen, *J. Comput. Chem.*, **17**, 940 (1996).
10. A. J. Hopfinger and R. A. Pearlstein, *J. Comput. Chem.*, **5**, 486 (1984).
11. J. M. Leonard and W. P. Ashman, *Comput. Chem.*, **11**, 952 (1990).
12. W. D. Cornell, P. Cieplak, C. I. Bayly, I. R. Gould, K. M. Merz Jr., D. M. Ferguson, D. C. Spellmeyer, T. Fox, J. W. Caldwell, and P. A. Kollman, *J. Am. Chem. Soc.*, **117**, 5179 (1995).

13. P.-O. Norrby, K. Wärnmark, B. Åkermark, and C. Moberg, *J. Comput. Chem.*, **16**, 620 (1995).
14. (a) K. Gundertofte, T. Liljefors, P.-O. Norrby, and I. Pettersson, *J. Comput. Chem.*, **17**, 429 (1996); I. Pettersson and T. Liljefors, In *Reviews in Computational Chemistry*, *Vol. 9*, K. B. Lipkowitz and D. B. Boyd, Eds., VCH, New York, 1996, p. 167. (b) Note that the errors for CFF reported in (a) refer to calculations using both high- and low-quality parameters. Counting only results obtained with parameters developed according to ref. 8, CFF is in fact statistically indistinguishable from the best force fields in the study.
15. F. Mohamadi, N. G. J. Richards, W. C. Guida, R. Liskamp, M. Lipton, C. Caulfield, G. Chang, T. Hendrickson, and W. C. Still, *J. Comput. Chem.*, **11**, 440 (1990). MacroModel, version 5.5.
16. For exact file formats, see the BatchMin manual, which accompanies MacroModel (ref. 15).
17. C. Bayly, P. Cieplak, W. Cornell, and P. A. Kollman, *J. Phys. Chem.*, **97**, 10269 (1993).
18. C. M. Breneman and K. B. Wiberg, *J. Comput. Chem.*, **11**, 361 (1990).
19. D. M. Schnur, M. V. Grieshaber, and J. P. Bowen, *J. Comput. Chem.*, **11**, 361 (1990). N. L. Allinger, X. Zhou, and J. Bergsma, *J. Mol. Struct. (Theochem)*, **312**, 69 (1994).
20. W. H. Press, S. A. Teukolsy, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C, 2nd Ed.*, Cambridge University Press, New York, 1992.
21. For the symmetric, positive definite matrix **A**, the procedure is equivalent to a determination of eigenvectors and eigenvalues. The regression based on the most significant eigenvectors is also equivalent to principal component regression (PCR).
22. M. J. Frisch, G. W. Trucks, H. B. Schlegel, P. M. W. Gill, B. G. Johnson, M. A. Robb, J. R. Cheeseman, T. Keith, G. A. Petersson, J. A. Montgomery, K. Raghavachari, M. A. Al-Laham, V. G. Zakrzewski, J. V. Ortiz, J. B. Foresman, J. Cioslowski, B. B. Stefanov, A. Nanayakkara, M. Challacombe, C. Y. Peng, P. Y. Ayala, W. Chen, M. W. Wong, J. L. Andres, E. S. Replogle, R. Gomperts, R. L. Martin, D. J. Fox, J. S. Binkley, D. J. Defrees, J. Baker, J. P. Stewart, M. Head-Gordon, C. Gonzalez, and J. A. Pople, GAUSSIAN-94, *Revision D.1*, Gaussian, Inc., Pittsburgh, PA, 1995.
23. A. D. Becke, *J. Chem. Phys.*, **98**, 5648 (1993); C. Lee, W. Yang, and R. G. Parr, *Phys. Rev. B*, **37**, 785 (1988).
24. (a) J. B. Foresman and M. J. Frisch, *Exploring Chemistry with Electronic Structure Methods, 2nd Ed.*, Gaussian, Inc., Pittsburgh, PA, 1996. M. J. Frisch, presentation at the 213th ACS National Meeting, San Francisco, CA, 1997. (b) D. A. Singleton, S. R. Merrigan, J. Liu, and K. N. Houk, *J. Am. Chem. Soc.*, **119**, 3385 (1997), and references cited therein.
25. L. S. Bartell and H. K. Higginbotham, *J. Chem. Phys.*, **42**, 851 (1965).
26. E. Hirota, H. Endo, S. Saito, and J. L. Duncan, *J. Mol. Spectrosc.*, **89**, 285 (1981).
27. MacMimic/MM3(92), InStar Software AB, IDEON Research park, S-223 70 Lund, Sweden.
28. A. P. Scott and L. Radom, *J. Phys. Chem.*, **100**, 16502 (1996).
29. The procedures are available as SGI mips2 executables (tested on Indigo^2 and Power Indigo^2, under Irix 5.3 and 6.2), or as source code. Numerical recipes procedures are not included in the distributed source code, but are included with permission in the executable code. Also available are several sets of Unix scripts, and an example ethane parameterization. E-mail to Per-Ola Norrby, peon@medchem.dfh.dk, for ftp download directions.
30. MM3 shortens the bonds to hydrogens to 92.7% for the calculation of van der Waals forces only, whereas MM3* uses the offset position for all calculations.